

## In the Claims

1 **1. (Currently Amended)** A processor-readable medium comprising  
2  
3 processor-executable instructions for:  
4

5 parsing an input file to recognize a file format of the input file, wherein the  
6 parsing repeatedly parses once with each of a plurality of component  
7 parsers contained within a compound parser, wherein ~~the compound~~  
8 ~~parser is extensible and~~ each of the plurality of component parsers is  
9 configured for recognition of a specific file format by which an input  
10 file is configured, wherein the compound parser is extensible, and  
11 wherein extending the compound parser comprises adding an  
12 additional component parser;

13 checking contents of the input file, according to the recognized file format,  
14 to determine whether executable code exists within the input file,  
15 wherein the checking comprises detecting executable code because  
16 its location within the input file is inconsistent with the recognized  
17 file format;

18 continuing to parse the input file until a component parser recognizes the  
19 file format of the input file ~~with all remaining component parsers~~  
20 ~~after at least one component parser recognizes the file format of the~~  
21 ~~input file;~~ and

22 sending a status in response to results of said checking, wherein sending a  
23 status comprises further instructions for:  
24  
25

1           sending a file-has-no-code status when the file format of the input  
2                     file was recognized and no executable code was found;  
3           sending a file-has-code status when executable code was found; and  
4           sending a don't-know status when the file format of the input file  
5                     was not recognized.

6  
7   **2-5. (Cancelled)**

8  
9   **6. (Original)** The processor-readable medium as recited in claim 1, wherein  
10       sending the status comprises further instructions for sending the status to an  
11       email program.

12  
13   **7. (Original)** The processor-readable medium as recited in claim 1, wherein  
14       sending the status comprises further instructions for sending the status to an  
15       instant messaging program.

16  
17   **8. (Original)** The processor-readable medium as recited in claim 1, wherein  
18       sending the status comprises further instructions for sending the status to an  
19       internet browsing program.

20  
21   **9-12. (Cancelled)**

22  
23   **13. (Currently Amended)** The processor-readable medium as recited in claim  
24       11, additionally comprising further instructions for continuing to parse the  
25

1 input file with all remaining component parsers after at least one  
2 component parser recognizes the file format of the input file ~~parsing the~~  
3 ~~input file until a component parser recognizes the file format of the input~~  
4 ~~file.~~

- 5
- 6 **14. (Currently Amended)** A method of detecting code-free files, comprising:
- 7 identifying a new file format, wherein ability to recognize the new file  
8 format is functionality to be extended to a compound parser;  
9 configuring a new component parser according to the new file format,  
10 wherein the new component parser is configured to recognize files  
11 of the new format and also to recognize executable code in files of  
12 the new format by locating executable code that is inconsistent with  
13 the new file format; and  
14 extending functionality of the compound parser by adding the new  
15 component parser to the compound parser;  
16 wherein the compound parser, having extended functionality, is configured  
17 to operate to parse an input file by:  
18 parsing an the input file with a the compound parser, wherein the  
19 compound parser is configured to include a plurality of  
20 component parsers, wherein each component parser is  
21 configured to recognize a specific data file format;  
22 analyzing contents of the input file according to the recognized  
23 specific file format, where available, to determine if the input  
24 file contains executable code; and
- 25

1 sending a status in response to results of said analyzing.

2  
3 **15. (Original)** The method as recited in claim 14, additionally comprising:  
4 sending a file-has-no-code status when the file format of the input file was  
5 recognized and no executable code was found; and  
6 sending a file-has-code status when executable code was found.

7  
8 **16. (Original)** The method as recited in claim 14, additionally comprising  
9 sending a don't-know status when a file format of the input file was not  
10 recognized.

11  
12 **17. (Original)** The method as recited in claim 14, additionally comprising  
13 sending the status to an email program.

14  
15 **18. (Original)** The method as recited in claim 14, additionally comprising  
16 sending the status to an instant messaging program.

17  
18 **19. (Original)** The method as recited in claim 14, additionally comprising  
19 sending the status to an internet browsing program.

20  
21 **20. (Original)** The method as recited in claim 14, wherein parsing the input file  
22 comprises parsing the input file with each of the plurality of component  
23 parsers within the compound parser.  
24  
25

1 **21. (Currently Amended)** An apparatus for detecting code-free files,  
2 comprising:

3 a compound parser configured to repeatedly parse an input file, wherein  
4 each component parser within the compound parser is configured to  
5 recognize executable code within a specific file format selected from  
6 among a group of data file formats; and

7 a controller to examine success of each of the component parsers to  
8 recognize the specific file format for which it was configured to  
9 recognize and to find executable code within the input file, wherein  
10 the controller is configured to send a status in response to results of  
11 said checking, wherein sending a status comprises:

12 sending a file-has-no-code status when the file format of the  
13 input file was recognized and no executable code was  
14 found;

15 sending a file-has-code status when executable code was  
16 found; and

17 sending a don't-know status when the file format of the input  
18 file was not recognized.

19  
20 **22. (Cancelled)**

21  
22 **23. (Original)** The apparatus as recited in claim 21, wherein the apparatus for  
23 detecting code-free files is additionally configured to send the status to an  
24 email program.  
25

1  
2 **24. (Original)** The apparatus as recited in claim 21, wherein the apparatus for  
3 detecting code-free files is additionally configured to send the status to an  
4 instant messaging program.

5  
6 **25. (Original)** The apparatus as recited in claim 21, wherein the apparatus for  
7 detecting code-free files is additionally configured to send the status to an  
8 internet browsing program.

9  
10 **26. (Original)** The apparatus as recited in claim 21, additionally configured to  
11 send the status to:  
12 a firewall;  
13 a host intrusion detector; or  
14 a host vulnerability assessor.

15  
16 **27. (Original)** The apparatus as recited in claim 21, additionally configured to  
17 send the status to a program selected from a group of programs,  
18 comprising:  
19 a backup program;  
20 a CD/DVD burning program; and  
21 a P2P file-sharing program.  
22  
23  
24  
25

1 **28. (Original)** The apparatus as recited in claim 21, wherein each of the  
2 component parsers is configured to recognize one of a plurality of data file  
3 formats.

4  
5 **29. (Original)** The apparatus as recited in claim 21, wherein the compound  
6 parser is configured to allow extension by addition of a new component  
7 parser to the compound parser, wherein the new component parser  
8 recognizes a further file format and recognizes executable code within the  
9 further file format.

10  
11 **30. (New)** The processor-readable medium as recited in claim 1, wherein  
12 adding an additional component parser comprises instructions for:  
13 identifying a new file format, wherein ability to recognize the new file  
14 format is functionality to be extended to the compound parser;  
15 configuring a new component parser according to the new file format,  
16 wherein the new component parser is configured to recognize files  
17 of the new format and also to recognize executable code in files of  
18 the new format by locating executable code that is inconsistent with  
19 the new file format; and  
20 extending functionality of the compound parser by adding the new  
21 component parser to the compound parser.

22  
23  
24  
25